

CLAIMS

What is claimed is:

1. A method for administering a serial bus, the bus facilitating communication between node devices connected to the bus and communicating over the bus in the form of packetized communication between node devices, wherein a first type of packet comprises asynchronous packets characterized by the absence of a requirement that an unarbitrated response or ack packet be sent in response to transmission of a packet of the first type, wherein a second type of packet comprises asynchronous packets, the method comprising:

if there is a packet of the second type to be sent, then concatenating the packet of the second type to a plurality of packets of the first type and sending the plurality of packets of the first type followed by the concatenated packet of the second type; and

if there is no packet of the second type to be sent, then concatenating a bogus ack packet to the plurality of packets of the first type and sending the plurality of packets of the first type followed by the concatenated bogus ack packet.

2. The method of claim 1, wherein concatenating the packet of the second type is performed by link hardware.

3. The method of claim 1, wherein concatenation of the bogus ack packet is performed by link hardware.

4. The method of claim 1, wherein concatenation of the bogus ack packet is performed by PHY hardware.

5. The method of claim 4, wherein link hardware is unaware that the PHY hardware performs concatenation.

6. The method of claim 1, further comprising inspecting a first quadlet of a packet to determine a packet type.

7. The method of claim 6, wherein the first quadlet contains a transaction code, further comprising:
 - determining from the transaction code that the packet is a stream packet; and
 - determining that transmission is not occurring during an isochronous period.
8. A method for administering a serial bus, the bus facilitating communication between node devices connected to the bus and communicating over the bus in the form of packetized communication between node devices, wherein a first type of packet comprises asynchronous packets characterized by the absence of a requirement that an unarbitrated response or ack packet be sent in response to transmission of a packet of the first type, the method comprising:
 - receiving a packet of the first type;
 - if there are additional packets of the first type to be sent, and if fly-by concatenation is permitted, then concatenating the additional packets of the first type to the received packet and sending the received packet and concatenated packets; and
 - if there are more packets of the first type to be sent and fly-by concatenation is not permitted then sending the received packet and arbitrating for control of the bus and sending the concatenated packets.
9. The method of claim 8, wherein concatenating the additional packets is performed by PHY hardware.
10. The method of claim 8, wherein arbitrating for control of the bus is performed by PHY hardware.
11. The method of claim 8, further comprising inspecting a first quadlet of a packet to determine a packet type.

12. The method of claim 11, wherein the first quadlet contains a transaction code, further comprising:

determining from the transaction code that the packet is a stream packet; and

determining that transmission is not occurring during an isochronous period..

13. A method for administering a serial bus, the bus facilitating communication between node devices connected to the bus and communicating over the bus in the form of packetized communication between node devices, wherein a first type of packet comprises asynchronous packets characterized by the absence of a requirement that an unarbitrated response or ack packet be sent in response to transmission of a packet of the first type, wherein a second type of packet comprises asynchronous packets, the method comprising:

receiving a packet of the first type;

if there is a packet of the second type to be sent, and if fly-by concatenation is permitted, then concatenate the packet of the second type to the received packet of the first type;

if there is a packet of the second type to be sent and fly-by concatenation is not permitted then send the received packet; arbitrate for the bus; and

send the packet of the second type.

14. The method of claim 13, wherein concatenating the packet is performed by PHY hardware.

15. The method of claim 13, wherein arbitrating for control of the bus is performed by PHY hardware.

16. The method of claim 13, further comprising inspecting a first quadlet of a packet to determine a packet type.

17. The method of claim 16, wherein PHY hardware determines the packet of the first type to be an asynchronous stream packet by reading a time code in the packet of the first type.

18. A method for administering a serial bus, the bus facilitating communication between node devices connected to the bus and communicating over the bus in the form of packetized communication between node devices, wherein a first type of packet comprises asynchronous packets characterized by the absence of a requirement that an unarbitrated response or ack packet be sent in response to transmission of a packet of the first type, wherein a second type of packet comprises asynchronous packets, the method comprising:

- receiving a packet of the first type;
- determining that there are no packets of the second type to be sent;
- if fly-by concatenation is permitted then concatenating a bogus ack packet to the received packet and sending the received packet and the bogus ack packet; and
- if fly-by concatenation is not permitted then sending the received packet, arbitrating for the bus, and sending a bogus ack packet.

19. The method of claim 18, wherein concatenating the bogus ack packet is performed by PHY hardware.

20. The method of claim 18, wherein arbitrating for control of the bus is performed by PHY hardware.

21. The method of claim 18, further comprising inspecting a first quadlet of a packet to determine a packet type.

22. The method of claim 21, wherein the first quadlet contains a transaction code, further comprising:

- determining from the transaction code that the packet is a stream packet; and

determining that transmission is not occurring during an isochronous period.

23. A computer program product, containing instructions which, when executed by a computer, administer a serial bus that facilitates communication between node devices connected to the bus and communicating over the bus in the form of packetized communication between node devices, wherein a first type of packet comprises asynchronous packets characterized by the absence of a requirement that an unarbitrated response or ack packet be sent in response to transmission of a packet of the first type, wherein a second type of packet comprises asynchronous packets, by performing the acts of:

if there is a packet of the second type to be sent, then concatenating the packet of the second type to a plurality of packets of the first type and sending the plurality of packets of the first type followed by the concatenated packet of the second type; and

if there is no packet of the second type to be sent, then concatenating a bogus ack packet to the plurality of packets of the first type and sending the plurality of packets of the first type followed by the concatenated bogus ack packet.

24. A computer program product, containing instructions which, when executed by a computer, administer a serial bus that facilitates communication between node devices connected to the bus and communicating over the bus in the form of packetized communication between node devices, wherein a first type of packet comprises asynchronous packets characterized by the absence of a requirement that an unarbitrated response or ack packet be sent in response to transmission of a packet of the first type, wherein a second type of packet comprises asynchronous packets, by performing the acts of:

receiving a packet of the first type;

if there are additional packets of the first type to be sent, and if fly-by concatenation is permitted, then concatenating the additional packets of the first type to the received packet and sending the received packet and concatenated packets; and

if there are more packets of the first type to be sent and fly-by concatenation is not permitted then sending the received packet and arbitrating for control of the bus and sending the concatenated packets.

25. A computer program product, containing instructions which, when executed by a computer, administer a serial bus that facilitates communication between node devices connected to the bus and communicating over the bus in the form of packetized communication between node devices, wherein a first type of packet comprises asynchronous packets characterized by the absence of a requirement that an unarbitrated response or ack packet be sent in response to transmission of a packet of the first type, wherein a second type of packet comprises asynchronous packets, by performing the acts of:

receiving a packet of the first type;

if there is a packet of the second type to be sent, and if fly-by concatenation is permitted, then concatenate the packet of the second type to the received packet of the first type;

if there is a packet of the second type to be sent and fly-by concatenation is not permitted then send the received packet; arbitrate for the bus; and

send the packet of the second type.

26. A computer program product, containing instructions which, when executed by a computer, administer a serial bus that facilitates communication between node devices connected to the bus and communicating over the bus in the form of packetized communication between node devices, wherein a first type of packet comprises asynchronous packets characterized by the absence of a requirement that an unarbitrated response or ack packet be sent in response to transmission of a packet of the first type, wherein a second type of packet comprises asynchronous packets, by performing the acts of:

receiving a packet of the first type;

determining that there are no packets of the second type to be sent;

if fly-by concatenation is permitted then concatenating a bogus ack packet to the received packet and sending the received packet and the bogus ack packet; and
if fly-by concatenation is not permitted then sending the received packet, arbitrating for the bus, and sending a bogus ack packet.

27. A device connected to a serial bus, the device containing instructions which, when executed by a computer, administer a serial bus that facilitates communication between node devices connected to the bus and communicating over the bus in the form of packetized communication between node devices, wherein a first type of packet comprises asynchronous packets characterized by the absence of a requirement that an unarbitrated response or ack packet be sent in response to transmission of a packet of the first type, wherein a second type of packet comprises asynchronous packets, by performing the acts of:

if there is a packet of the second type to be sent, then concatenating the packet of the second type to a plurality of packets of the first type and sending the plurality of packets of the first type followed by the concatenated packet of the second type; and

if there is no packet of the second type to be sent, then concatenating a bogus ack packet to the plurality of packets of the first type and sending the plurality of packets of the first type followed by the concatenated bogus ack packet.

28. A device connected to a serial bus, the device containing instructions which, when executed by a computer, administer a serial bus that facilitates communication between node devices connected to the bus and communicating over the bus in the form of packetized communication between node devices, wherein a first type of packet comprises asynchronous packets characterized by the absence of a requirement that an unarbitrated response or ack packet be sent in response to transmission of a packet of the first type, wherein a second type of packet comprises asynchronous packets, by performing the acts of:

receiving a packet of the first type;

if there are additional packets of the first type to be sent, and if fly-by concatenation is permitted, then concatenating the additional packets of the first type to the received packet and sending the received packet and concatenated packets; and

if there are more packets of the first type to be sent and fly-by concatenation is not permitted then sending the received packet and arbitrating for control of the bus and sending the concatenated packets.

29. A device connected to a serial bus, the device containing instructions which, when executed by a computer, administer a serial bus that facilitates communication between node devices connected to the bus and communicating over the bus in the form of packetized communication between node devices, wherein a first type of packet comprises asynchronous packets characterized by the absence of a requirement that an unarbitrated response or ack packet be sent in response to transmission of a packet of the first type, wherein a second type of packet comprises asynchronous packets, by performing the acts of:

receiving a packet of the first type;

if there is a packet of the second type to be sent, and if fly-by concatenation is permitted, then concatenate the packet of the second type to the received packet of the first type;

if there is a packet of the second type to be sent and fly-by concatenation is not permitted then send the received packet; arbitrate for the bus; and

send the packet of the second type.

30. A device connected to a serial bus, the device containing instructions which, when executed by a computer, administer a serial bus that facilitates communication between node devices connected to the bus and communicating over the bus in the form of packetized communication between node devices, wherein a first type of packet comprises asynchronous packets characterized by the absence of a requirement that an unarbitrated response or ack packet be sent in response to transmission of a packet of the first type, wherein a second type of packet comprises asynchronous packets, by performing the acts of:

receiving a packet of the first type;
determining that there are no packets of the second type to be sent;
if fly-by concatenation is permitted then concatenating a bogus ack packet to the received packet and sending the received packet and the bogus ack packet; and
if fly-by concatenation is not permitted then sending the received packet, arbitrating for the bus, and sending a bogus ack packet.